Recognizing Complex Events using Large Margin Joint Low-Level Event Model

Hamid Izadinia, and Mubarak Shah

Computer Vision Lab, University of Central Florida, Orlando, Florida
{izadinia,shah}@eecs.ucf.edu

Abstract. In this paper we address the challenging problem of complex event recognition by using low-level events. In this problem, each complex event is captured by a long video in which several low-level events happen. The dataset contains several videos and due to the large number of videos and complexity of the events, the available annotation for the low-level events is very noisy which makes the detection task even more challenging. To tackle these problems we model the joint relationship between the low-level events in a graph where we consider a node for each low-level event and whenever there is a correlation between two low-level events the graph has an edge between the corresponding nodes. In addition, for decreasing the effect of weak and/or irrelevant low-level event detectors we consider the presence/absence of low-level events as hidden variables and learn a discriminative model by using latent SVM formulation. Using our learned model for the complex event recognition, we can also apply it for improving the detection of the low-level events in video clips which enables us to discover a conceptual description of the video. Thus our model can do complex event recognition and explain a video in terms of low-level events in a single framework. We have evaluated our proposed method over the most challenging multimedia event detection dataset. The experimental results reveals that the proposed method performs well compared to the baseline method. Further, our results of conceptual description of video shows that our model is learned quite well to handle the noisy annotation and surpass the low-level event detectors which are directly trained on the raw features.

1 Introduction

The majority of current human action recognition work deals with the classification of short video clips (e.g. 3-10 sec) which contain some simple and well-defined actions such as running, biking, diving, etc, and the main challenges are how to deal with low resolution, arbitrary camera motion, occlusion and clutter in the scene. However, real lifetime videos are of longer length which contain complex events happening at specific place and time such as birthday party and wedding ceremony; such videos may depict complex scenes and involve a number of human actions in which people interact with each other and/or with objects. For example a video of *birthday party* event can be described by the objects (*cake, candle*), scene (*indoor, outdoor*), actions (*person singing, laughing*) and

Fig. 1. Examples from complex video event categories: (from left to right, column wise) *boarding trick, feeding animal, landing fish, wedding, woodworking project, birthday party, changing tire, flash mob, vehicle unstuck, grooming animal, making sandwich, parade, parkour, repairing appliance, sewing project.*

surrounding voices (*music, cheering*) that happen in it. Therefore, it is apparent that classifying a complex realistic event is a much more challenging task than just recognizing a set of motion discriminative actions (low level events) in standard datasets (such as KTH [1], UCF-Sports [2], UCF50 [3], and HMDB [4]). Some example video frames from complex event categories considered in this paper are shown in Fig. 1.

Recently, the bag-of-words (BoW) approach has achieved impressive results in many recognition problems including action recognition [5, 6]. However, this approach has innate limitations in representation and semantic description of the underlying data as it jumps directly from low level features to the very high level class labels. Therefore, the methods which are based on BoW approach cannot easily provide any semantic intermediate description of the data.

For recognizing complex events, we argue that it is crucial to learn the low-level events along with their relationships to the event categories. For example, for *Birthday party* event, low-level events may include: *person cheering, person singing, person blowing candles, person taking pictures*, etc. For each low-level event we use a collection of various features to learn its model. We then use the learned low-level event detectors to train a discriminative model for recognizing complex events. To this end, we model the joint relations between the low-level events by a latent graphical model. In our model, we have a node for each low-level event and the edges between the nodes represent the correlations between the low-level events. Since, the number of all possible co-occurrence of these low-level events is very large, we take the advantage of the fact that a large portion of possible co-occurrences is rather unlikely to happen and exploit only those which have high rate of coincidence. We consider the presence or absence of low-level events as latent variables and learn their correlations in a latent SVM framework, which simultaneously alleviate the problem of noisy low-level event detectors and improves the accuracy of high-level event recognition.

The overview of the proposed method is summarized in Fig. 2. At the first stage the raw features extracted from the training videos along with the information obtained by low-level event annotation are used to train the low-level event detectors. The graph of low-level event co-occurrence is also constructed using

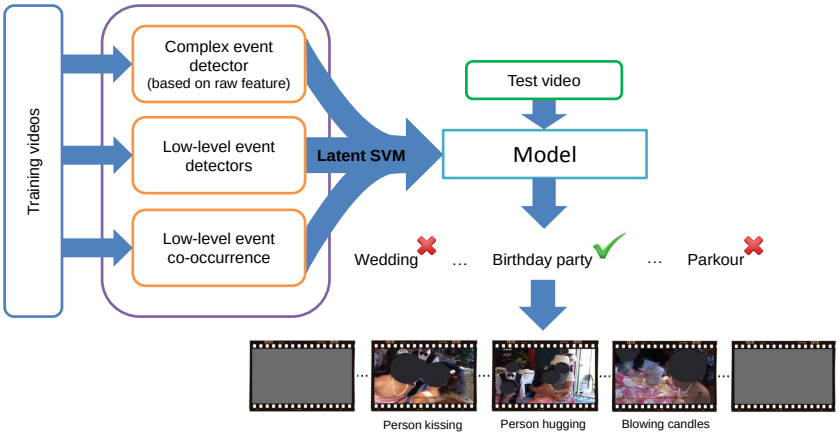

Fig. 2. Given the training videos and low-level event annotation, we train low-level event detectors and high-level event detector using raw feature. Then we employ co-occurrence of low-level events along with the individual low-level event detector outputs in latent SVM framework to detect the high-level event label. We also use the latent parameter vector for describing an unseen test video in terms of low-level events. For example, the given video of birthday party is described by the sequence of low-level events: *person kissing, person hugging and blowing candles*.

annotations. In addition, high level event detectors are trained using raw features directly. The final model is generated using the low-level events, co-occurrence graph and high-level event detectors. Our training data includes long sequences of each of 15 complex events which are divided into short clips of typically 10 seconds. Each short clip potentially contains one of 62 low-level events. Each clip is assigned to one of the 62 low-level event labels by human annotators, which are only used for training the detectors. At the testing time, we need to predict the category of a given complex event video. Thus, we use a latent SVM model in which the low-level event are treated as latent variables. Also, in our latent SVM framework, we learn the co-occurrence pattern of the low-level events for further improvement of the recognition performance. As an example, a given test video could be a short movie of a wedding ceremony that contains low-level events such as *kissing, hugging, dancing, taking picture*, at different temporal locations in a video. Using trained low-level event detectors, we can compute the confidence scores for the presence of the low-level events in all the 10 second clips of the test video. With our trained latent SVM model and the obtained confidence scores, we can accurately describe each video.

The key contributions of our work are as follows: First, our proposed model shows that learning low-level events can improve the recognition rate of complex events. Here, we model low-level events in a latent graphical model where for discovering the joint relations between low-level event a latent SVM is trained. Second, our model provides a flexible framework for using the combination of

various types of low-level features for modeling contextual information, local appearance, motion patterns and audio properties. Third, using trained latent SVM model, we can provide a semantic description of a given video which can be used in problems like video retrieval, where the aim is to detect the presence or absence of a semantic concept in video.

2 Related Work

The explosive growth of digital videos on the Internet has made an urgent necessity for having efficient methods for video analysis. Amongst all, high level video event classification and recognition is one of the most critical problems that should be solved to this end. While the action recognition problem, which can be considered as low-level event recognition, is widely explored, the problem of event recognition is not much explored [7–9]. The challenging nature of event recognition problem lies in the fact that simple actions are the building blocks of events while the action recognition problem is itself one of the most challenging recognition problems to date. Thus, we argue that it is very logical to treat the action recognition as an intermediate step in recognizing complex events.

On the other hand, the use of different attributes for the recognition task has recently been explored in different computer vision applications such as object classification [10–13], image ranking and retrieval [14] and human action recognition [15]. Some of the attributes that has been used in these methods have semantic meaning while some of them are data driven attributes[15]. The data driven attributes are extracted from training data based on raw features. These attributes can only increase the performance of the recognition but do not provide any conceptual description about the content of the video.

Our notion of low-level events is similar to the attributes in the sense that both are used as a source of intermediate information for recognition of a more complex task. However, in the literature, an attribute refers to an atomic part of a more general category while each of our low-level events is itself a general category. Thus, the general notion of attribute stands at a smaller granularity than our low-level events. For example in the object recognition a set of possible attributes for recognizing objects can be (furry, leg, metallic surfaces, 3D boxy) [10] and in action recognition can be (up-down motion, torso motion, twist) [15]. Whereas, some of our low-level events are (Person dancing, people marching, animal eating). The other main difference of our approach with the attribute based methods is that, in the attribute based methods, the presence/absence of the attributes is used to improve the recognition task, but there is no concrete representation for each of the attributes and thus the attribute detection is not that informative. Whereas, each of our low-level events refer to a certain clip and our method learns the low-level event for both event recognition and temporal video description. Recently, [16–18] modeled the temporal structure of the video. However, they anchor a predefined number of low-level events/actions in temporal domain and attempt to find the best discriminative temporal model for each high-level event/action. In our work we do not impose

any constraint on the temporal location of each low-level event but instead we learn the co-occurrence pattern of the low-level events for further improvement of the recognition performance. Thus, we are not limited by any kind of prior information about the temporal locations of low-level events and learn the co-occurrence via a latent SVM framework.

3 Complex Event Recognition using Low-Level Events

For classifying videos we start by considering each video as a collection of low-level events. Each low-level event can either refer to a simple action that is performed by one or more actors such as *person walking*, a complex action that takes place while interacting with other objects (*person petting*) or a particular behavior that is performed by a group of people (*people dancing*). Thus, for solving the video classification we propose to learn low-level events along with their correlations by analyzing the video sequence temporally and using a set of diverse features: ISA (independent subspace analysis) [5], STIP (spatio-temporal interest point descriptor) [19], Dollar [20], GIST [21], SIFT [22] and MFCC (Mel-frequency cepstral coefficients) [23] for describing each low-level event. The correlations between low-level events are then learned in latent SVM framework.

For learning the low-level events we have manually annotated the training videos, as is typically done in human action recognition work. Of course, we assume these labels are considered not to be available at the test time. For each of these low-level events a classifier is trained based on the low-level features.

Using the low-level event detectors, we then compute a feature vector for each event video and use it for training high level event detectors. To this end, we need to compute the confidence scores of different low-level event detectors for the clips of each video. The low-level events are of different temporal length, since the videos contain real world events. Thus, we compute the confidence score of the low level detectors on overlapping clips of the video in a hierarchical fashion. At the first level of the hierarchy the confidence scores are computed using fixed length overlapping clips, then at each higher level the confidence score for two adjacent clips of the lower level is computed. After computing all the confidence feature vectors, the final high-level feature vector for the video is computed by max pooling over all confidence vectors.

3.1 Large Margin Learning based on underlying Latent Structure

In this section, we address the problem of learning a model for labeled and structured data. For the high level event recognition problem considered in this paper, we explore the underlying structure based on a joint relation graph which is constructed using the co-occurrence of the low-level events.

Each training sample is represented by (x, z, y) in which x is a video and $y \in \mathcal{Y}$ denotes its class label. And the low-level event representation of a video is defined by a C-dimensional binary vector $z = (z_1, ..., z_C)$ where each dimension shows the presence/absence of a specific low-level event in a video. For instance,

if the ith video belongs to the *Birthday party* event and the cth dimension corresponds to the *Person lighting candle* low-level event, z_c would probably be equal to 1.

We consider a training set that consists of n input/output pairs $(x_1, y_1), ..., (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$. Given the training data, we are interested in learning a discriminative function $\mathcal{F}_\theta : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ over the feature vector of a video and its event class label. Here $\mathcal{F}$ is parameterized by θ. During testing, we can predict the class label of a high-level event video using Eq. (1);

$$y^* = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \, \mathcal{F}_\theta(x, y). \tag{1}$$

Since we consider latent low-level event representation for each video, the discriminative function $\mathcal{F}$ scores based on the latent variable which is computed by $\mathcal{F}_\theta(x, y) = \max_z \ \Theta^\top \Phi(x, z, y)$. Here $\Theta^\top \Phi(x, z, y)$ depends on global event potential, unary low-level event potential and joint low-level event potential:

$$\Theta^\top \Phi(x, z, y) = \theta_y^\top \phi(x) + \sum_{j \in \mathcal{V}} (\theta_{z_j}^\top \varphi(x) + \beta_{(y,j)}) + \sum_{(j,k) \in \mathcal{E}} \theta_{(j,k)}^\top \psi(z_j, z_k), \tag{2}$$

in which $\Theta = (\theta_y, \theta_{z_j}, \theta_{(j,k)})$ is the parameter (weight) vector of $\mathcal{F}$. The potentials are defined in the following.

Global event potential: The global potential $\theta_y^\top \phi(x)$ represents a linear discriminative model for event detection without considering low-level events, where each video x is represented by a feature vector $\phi(x)$. In order to speed up the training process we pre-train a classifiers for each event and incorporate θ_y to regularize the confidence score of the event classifiers. Thus, without loss of generality $\phi(x)$ refers to the confidence score of the corresponding event classifier for the input video x. However, as we use different feature types (i.e. image, video and audio), we need to pre-train a classifier for each feature type so the score of each event classifier is weighted by θ_y that is a k dimensional vector for k different feature types.

Unary low-level event potential: The low-level event potential $(\theta_{z_j}^\top \varphi(x) + \beta_{(y,j)})$ determines the occurrence of each low-level event in a video. We can use the raw feature vector and then train a large parameter vector for recognizing each low-level event, but similar to the global potential, we use a pre-trained binary classifier for each low-level event. Therefore, the unary potential for each low-level event is the confidence score produced by each low-level event detector and $\beta_{(y,j)}$, which represents the occurrence of each low-level event in each event class.

Joint low-level event potential: There is a meaningful relationship in the co-occurrence of more than one low-level event in a video. For example, there are a certain number of low-level event e.g. *person kissing, taking picture, person dancing* which frequently occur in a particular event such as *wedding ceremony*, while it is very unlikely that some other low-level events like *person hammering* may occur in the same event. The joint potential $\theta_{(j,k)}^\top \psi(z_j, z_k)$ incorporates the co-occurrence of low-level events in training the classifier. Since we only consider

Fig. 3. The low-level events joint relation model computed by running maximum spanning tree on the complete co-occurrence graph. The weight of edges express the normalized co-occurrence between the vertices. The darker edges show stronger correlation between the low-level events.

presence and absence of low-level events as the latent variable, we have four possible joint potentials $\{(0,0),(0,1),(1,0),(1,1)\}$ between any two low-level events.

In practice, some low-level event pairs may have rather weak correlations, including both in their presence or absence. For example, the low-level event pairs (*person dancing* and *person using tire tube*), or (*person jumping*, *person drinking*) indeed do not have too much correlations, that is to say, the presence/absence of one low-level event will not contribute to that of another (i.e., their occurrence are independent of each other). Based on this observation, we remove the weaker relations and only consider the strongly correlated pairs. The selection of low-level events can be manually determined by experts or automatically selected by some data-driven approaches. In this paper, we measure the correlations of low-level event pair using the normalized co-occurrence defined by $\frac{\mathcal{N}(z_j,z_k)}{\mathcal{N}(z_j)\mathcal{N}(z_k)}$ in which $\mathcal{N}(.)$ and $\mathcal{N}(.,.)$ respectively count the number of occurrences and co-occurrences in the entire training set using annotations. Once we compute the concept pair co-occurrence, we construct the correlation graph in which the low-level events represent vertices and the weight of edges are the normalized co-occurrences. We cannot find the optimum low-level event representation over complete correlation graph without enumerating the entire set of

combinations which is exponential in cardinality of each node (i.e. $|\{0,1\}| = 2$ and for 62 low-level events is 2^{62}). To eliminate this problem, we compute the maximum spanning tree to find a co-occurrence tree so that only the most correlated pairs are adjacent. In this case, the inference problem becomes tractable and can be solved by dynamic programming. Fig. 3 shows the maximum spanning tree obtained for 62 low-level events. As shown in this figure, the connection between low-level event pairs are meaningful. For instance, *person surfing, person jumping and person sliding* are connected which are usually co-occur in *boarding trick* event. Another example is *person throwing* and *animal eating* which are usually co-occur in *feeding animal event* videos.

3.2 Large Margin Learning

We train a binary classifier for each complex event class. Each classifier scores an example x using Eq. 1, so we must learn the parameter vector Θ from the set of positive and negative samples. The parameter vector Θ for each event class is trained iteratively by minimizing the objective function

$$f(\Theta) = \frac{\lambda}{2}\|\Theta\|^2 + \sum_{i=1}^{n} R_i(\Theta), \tag{3}$$

where λ makes trade-off between generalization and the data fitting. The risk function R_i is computed based on the optimum latent variable z^* and the predicted class label y^* for each training sample. We define inference function $\mathcal{G}(x, z, y, \Theta) = \Theta^\top \Phi(x, z, y)$ which finds the optimum latent variables z^* based on the model parameter Θ using

$$z_y^* = \underset{z \in \mathcal{Z}}{\operatorname{argmax}}\ \mathcal{G}(x, z, y, \Theta) \quad \forall y \in \{-1, 1\}. \tag{4}$$

Then we use optimum latent variable z_y^* and find the predicted label for the ith video y^* by

$$y^* = \underset{y \in \{-1,1\}}{\operatorname{argmax}}\ (\mathcal{G}(x_i, z_y^*, y, \Theta) + \Delta(y, y_i)), \tag{5}$$

where y_i is the ground truth label and $\Delta(y, y_i)$ is the loss function. A variety of loss functions have been used in the literature, here we use $0/1$ loss function which is $\Delta(y, y_i) = 1$ if $y \neq y_i$, and $\Delta(y, y_i) = 0$ otherwise. Once the y^* is computed for the ith sample, the risk is computed by

$$R_i = \mathcal{G}(x_i, z_{y^*}^*, y^*, \Theta) + \Delta(y^*, y_i) - \mathcal{G}(x_i, z_{y_i}^*, y_i, \Theta). \tag{6}$$

Apparently, the risk function is non-zero if $y^* \neq y_i$. We minimize the objective function $f(\Theta)$ using non-convex regularized bundle method [24]. This method relies on the cutting plane technique, where a cutting plane in defined using the sub-gradient of objective function $f(\Theta)$ by

$$\delta_\Theta f = \lambda\Theta + \sum_{i=1}^{n} \left(\Phi(x_i, z_{y^*}^*, y^*) - \Phi(x_i, z_{y_i}^*, y_i)\right). \tag{7}$$

Low-level event	ISA	STIP	Dollar	SIFT	GIST	MFCC	Low-level event	ISA	STIP	Dollar	SIFT	GIST	MFCC
Person surfing	61.6	37.9	2.3	40.7	25.8	2.4	Person laughing	2.8	3.0	1.2	11.8	1.1	1.8
People marching	48.4	55.4	23.7	53.4	25.5	25.3	Lighting candle	11.1	0.4	0.3	0.3	0.3	0.2
Person carving	49.6	43.2	8.8	45.6	18.7	53.3	Person squatting	2.5	1.5	1.4	7.3	2.0	10.8
Person sewing	49.9	19.4	24.7	19.6	12.2	23.8	Person hugging	5.2	8.9	3.6	10.8	1.4	1.8
Vehicle moving	42.3	47.6	14.3	29.0	26.9	15.3	Wheel rotating	2.4	10.4	1.4	10.7	1.0	1.0
Animal eating	24.4	23.8	11.2	44.7	7.0	16.7	Using tire tube	10.4	5.3	4.0	7.5	4.0	4.9
People dancing	31.2	42.7	13.2	34.3	7.9	3.7	Person drilling	6.3	5.7	1.6	7.8	10.3	1.1
Person singing	30.8	34.8	7.8	34.7	6.0	40.2	Person falling	6.8	9.8	3.0	6.6	3.2	4.3
Person washing	38.8	21.7	5.0	40.0	10.9	8.2	Person running	9.4	7.5	1.5	3.2	1.3	3.3
Person pointing	22.5	7.9	7.4	7.7	1.5	30.0	Person waving	5.8	3.2	2.3	8.7	1.6	2.5
Person kissing	29.0	12.7	6.3	8.2	1.9	10.3	Taking pictures	4.1	8.1	6.3	5.0	2.2	3.0
Person sliding	26.7	14.9	4.6	18.9	16.0	3.0	Blowing candles	4.7	7.0	2.0	7.6	1.6	1.9
Open door	26.6	18.8	10.3	18.8	3.1	8.2	Person clapping	4.9	3.5	2.7	7.2	2.2	3.9
Turning wrench	23.1	17.9	4.7	26.1	5.3	13.5	Person casting	6.3	2.8	1.0	3.8	0.7	0.9
Person reeling	25.1	10.6	2.2	14.7	12.3	2.2	Person petting	6.0	1.4	0.7	1.8	0.7	3.8
Person planing	16.8	14.7	9.2	22.8	15.8	8.2	Person wiping	5.7	0.6	0.4	1.8	0.3	0.8
Person jumping	17.7	20.5	12.3	21.6	11.1	21.1	Person bending	5.4	2.8	1.8	5.4	1.9	2.2
Person flipping	18.1	21.4	7.1	21.1	14.7	8.1	Person rolling	0.7	2.0	0.7	4.6	0.3	2.6
Person walking	13.5	19.2	10.5	21.1	9.9	6.0	Person climbing	3.6	4.0	1.8	1.8	0.8	2.0
Person cutting	9.1	3.4	2.9	20.6	2.1	3.1	Shake	3.7	0.3	0.5	0.6	0.3	0.4
Person dancing	8.9	18.0	3.4	19.6	4.5	3.4	Playing instrument	0.5	2.8	0.4	1.4	0.3	0.5
Spreading cream	19.0	16.1	3.7	8.5	2.5	5.4	Stir	2.0	2.7	0.4	0.4	0.3	1.3
Person eating	5.7	4.8	3.5	16.6	2.2	3.7	Person jacking car	1.6	2.7	1.1	1.5	0.6	0.7
Open box	1.0	6.6	0.3	16.1	0.3	0.7	Person cheering	0.8	1.6	0.6	1.5	0.7	2.6
Person throwing	15.5	5.5	1.7	9.5	0.9	2.4	Person cutting cake	2.2	0.8	1.1	0.9	0.4	0.6
Person hammering	4.0	12.2	8.6	15.2	6.4	4.8	Person pushing	1.4	1.0	0.8	2.1	0.8	0.7
Person using knife	11.8	14.7	11.4	7.6	2.1	5.1	Person polishing	1.9	1.3	1.0	1.2	0.6	1.7
Person sawing	7.1	2.9	4.0	5.7	6.0	14.5	Animal approaching	1.1	1.3	0.7	1.8	0.9	0.9
Fitting bolts	13.8	13.2	2.7	14.3	5.1	14.1	Person cleaning	1.5	0.8	0.9	1.5	0.4	0.7
Cutting fabric	13.8	1.2	3.2	11.4	0.7	10.6	Person drinking	1.3	0.7	0.9	0.5	0.4	0.5
Person writing	11.9	9.0	4.1	12.4	6.5	6.6	Person pouring	0.6	0.5	0.6	0.8	0.5	0.8

Table 1. The Average Precision of low-level event detection using different features.

Feature	ISA	STIP	Dollar	SIFT	GIST	MFCC
mean AP	13.56	11.24	4.54	13.04	5.08	7.07

Table 2. The mean average precision value using different features.

The bundle method iteratively builds an increasingly accurate piecewise quadratic lower bound of the objective function by selecting the most violated sample and building the bundle using the sub-gradient at that point. Such a cutting plane is a linear lower bound of the risk function $R(\Theta)$ and is a quadratic lower bound of the objective function $f(\Theta)$.

4 Experiments

To evaluate the performance of the proposed method, we present results for event recognition on the TRECVID11-MED event kit [25] which is the most challenging multimedia event dataset. This dataset contains 2,061 multimedia videos (i.e., video clips including both video and audio) collected from Internet. The videos are divided into 15 different events: *Boarding trick, Feeding animal, Landing fish, Wedding, Wood working project, Birthday party, Changing tire, Flash mob, Vehicle unstuck, Grooming animal, Making sandwich, Parade, Parkour, Repairing appliance*, and *Sewing project.* As the dataset contains plenty of videos, we randomly split the videos of each class in the dataset into 70% videos for training and 30% for testing and report the recognition rate using the precision criteria. For quantitative comparison we use Average Precision (AP) which is used in PASCAL VOC challenge [26]. The AP summarizes the characteristic of precision/recall curve, and is defined as the mean precision at a set of equally spaced recall levels $[0, 0.1, ..., 1]$. For a given class, the precision/recall curve is computed using the output confidence scores.

4.1 Feature Representation

We use six different feature types: ISA, Dollar and STIP as motion features; SIFT and GIST for local and global image appearance features, respectively. We also use MFCC along with its first and second derivatives as audio features. For ISA feature we use pre-trained convolutional ISA network which is provided in released package[1]. The Dollar descriptors are extracted around spatio-temporal interest points where a predefined space-time filter has significant response. For STIP feature we use 3D Harris corner detector and combination of HoG-HoF is used as a descriptor. For extracting SIFT and GIST features, we uniformly sample every K frame of each video and extract 128-D SIFT and 960-D GIST descriptors from each of those key frames. We also use a standard set of short-term MFCC features from down-sampled audio signal to 16kHz. We extract MFCC features from each frame of 25 ms with 10 ms overlap, and retain 21 coefficients as audio features.

4.2 Low-Level Event Detection

Table 1 shows the performance of our low-level event detectors using different types of features. This figure shows that for some of the low-level events the performance is very low which is due to lack of sufficient training samples and diverse patterns of low-level events appearing in the training video clips.

In addition, the average performance using each feature is summarized in Table 2. Although this table shows that ISA and SIFT had the highest average performance, Table 1 shows that each of the above features has the highest performance for some of the low-level events, when used separately. For example, the *MFCC* features obtains the highest average precision compared to other features in *singing* and *Person carving* low-level events, where the audio contains discriminative information. Whereas in motion dominant low-level events like *People marching* and *People dancing* the STIP features have higher accuracy. Thus, the need for using different feature types in a unified framework is obvious.

4.3 Complex Event Recognition

Fig. 4 demonstrates the unary part of the trained parameter vector θ_{z_j}. This figure shows the importance of individual low-level event detectors and that the relevant low-level event have higher weights. For example, in the *making sandwich* event, *person eating*, *person using knife* and *spreading cream* have the highest weights. Fig. 5 demonstrates the learned underlying structure for the *Birthday party* event. The edges are bolder whenever the corresponding learned pairwise correlation is of more importance. As expected, the latent learning procedure was successfully able to assign larger weights for (*open box*, *person singing*) and (*blowing candle*, *person eating*) edges, which quite frequently happen in a birthday party. While, the rarely co-occurring low-level event pairs like

[1] `http://ai.stanford.edu/~wzou/`

High-level event	ISA	STIP	Dollar	SIFT	GIST	MFCC	Linear SVM ensemble	Joint (LL event)	Joint (HL+LL)
Flash mob	62.7	60.7	80.8	78.3	72.9	78.5	85.9	88.8	91.9
Repairing appliance	77.6	63.2	63.8	57.9	49.0	70.2	80.8	73.5	78.2
Birthday party	63.2	28.2	47.6	35.3	20.2	59.0	70.9	76.0	78.2
Boarding trick	49.4	58.1	52.4	54.3	54.8	65.3	75.6	68.8	75.7
Landing fish	29.1	46.2	69.8	39.8	36.0	64.6	74.1	71.6	72.2
Parade	42.3	36.7	46.3	45.2	36.0	42.2	65.7	71.0	72.4
Vehicle unstuck	35.3	39.5	48.2	48.2	39.5	44.1	66.1	67.8	69.1
Parkour	27.1	34.1	67.8	35.4	43.8	62.0	53.4	65.3	66.4
Wedding	53.4	52.1	66.3	63.2	62.2	66.5	66.5	64.4	67.5
Woodworking project	45.8	24.1	47.3	31.9	30.8	55.9	57.6	64.8	65.3
Feeding animal	34.3	28.6	39.1	27.5	30.1	51.4	58.2	57.8	56.5
Sewing project	37.8	20.6	35.1	32.7	23.0	55.3	56.9	56.4	57.5
Grooming animal	24.9	27.7	36.2	28.8	28.3	49.7	45.7	48.0	51.0
Changing tire	20.3	7.6	29.5	19.1	17.4	45.0	46.5	48.1	47.7
Making sandwich	25.4	21.9	32.5	19.0	19.6	28.5	35.6	41.5	41.9
mean AP	55.87	37.57	41.12	50.85	36.63	41.90	62.63	**64.25**	**66.10**

Table 3. The average precision of our approach compared with the baseline methods. The first 6 columns show the results obtained using bag of words approach employing individual features. The next column shows the results obtained by training a linear SVM on the confidences of low-level and high-level event detectors, mean AP is better than the ones obtained by using any individual features. Following that under Joint LL event column we show the results obtained by joint relationship of LL using latent SVM, the performance is further improved here. Finally, in the last column we show results obtained using both high-level and low-level event detectors joint model trained using latent SVM, which provides the best results.

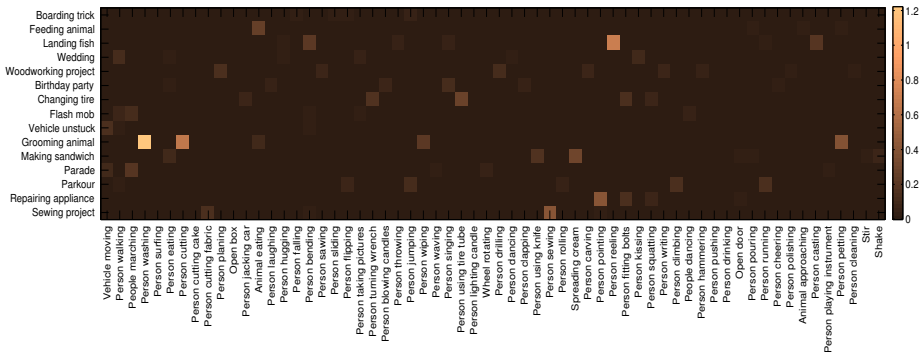

Fig. 4. The visualization of unary model parameters in terms of low-level events for the trained latent model of all events. The higher value shows more influence of low-level events in complex event recognition.

person walking and *person bending* are assigned low weights. On the other hand, a low pairwise weight is assigned to the low-level event *person cutting cake* which usually takes place in a *birthday party*. This is due to the noisy patterns of the *cutting cake* in the training videos and low performance of the *person cutting cake* detector. This reveals that the latent model could compensate the effect of noisy low-level event detectors by assigning a small value to the corresponding pairwise weights.

The classification results of our proposed method compared to the state of the art methods are summarized in Table 3. The best performance of the bag of words is obtained by using ISA features which is 55.87%. By fusing output of low-level event detectors with high-level event detectors for all feature types the performance is increased up to 62.63%. While co-occurrence of low-level events help remove the effect of noisy low-level event detectors and resulted in

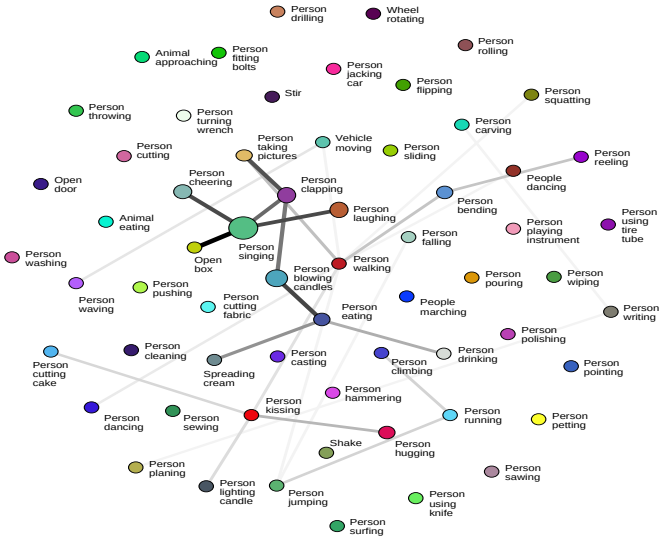

Fig. 5. The low-level event joint model trained by proposed latent method for *Birthday party*. The darker edge shows more discriminative joint for classifying this specific event.

64.25% average precision. Our proposed latent model using both low-level event and high-level event detectors has gained the highest performance i.e. 66.10%. Table 3 shows the comparison of classifier performance for each individual event. As can be observed, the precision of the latent event detector is higher than the other methods in most of the events. This is mostly visible in the *Flash mob*, *Birthday party* and *Parade* events which is due to their well performing low-level event detectors such as *People dancing*, *Person singing* and *People marching*.

4.4 Describing Video in terms of Low-Level Events

We want to label each clip (10 sec) of a given video with one of our low-level events. One simple approach for doing this is to directly use the output of low-level event detectors. However, as shown in Fig. 6 the low-level event detectors are too noisy due to errors in the human annotations. However, as shown in Fig. 4 our unary term parameter vector θ_z that are trained in the latent training procedure, can filter out irrelevant low-level events by assigning smaller weights to them. Therefore, for labeling each clip of a given video, we compute its confidence scores for all the low-level events. Having the vector of confidence scores, we simply compute $\theta_z^\top \varphi(x)$ and report the first five low-level events with highest $\theta_z^\top \varphi(x)$ value. The results obtained by this approach are shown in Fig. 6 for two sample videos. The caption of the videos contains the results obtained by the direct use of low-level event confidence scores and our approach.

Fig. 6. Temporal description of our method compared with the confidence score of low-level event detectors (LL confidence score) for two sample event videos. We sort the confidence score of all low-level events for each 10-second clip and show top five low-level events for each clip. The irrelevant low-level events with high confidence score are shown in bold.

5 Conclusion

In this paper we presented an event detection method based on latent low-level event model. Our proposed model learns a set of low-level event detectors and gets help from the low-level event co-occurrence in a latent SVM training procedure. Our model has the ability to filter out the noisy output of low-level event detectors and thus gains a good generalization for detecting low-level events. Additionally, our proposed method has the flexibility to get the benefits of using a set of different features in a unified framework. We evaluated the performance of our proposed method on the very challenging dataset and obtained impressive results on both event recognition and low-level event description.

Acknowledgements. The research presented in this paper is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20071. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

1. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: ICPR. (2004)

2. Rodriguez, M., Ahmed, J., Shah, M.: Action mach a spatio-temporal maximum average correlation height filter for action recognition. In: CVPR. (2008)
3. : Ucf50 action dataset. "http://vision.eecs.ucf.edu/data/UCF50.rar"
4. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: ICCV. (2011)
5. Le, Q., Zou, W., Yeung, S., Ng, A.: Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In: CVPR. (2011)
6. Wang, H., Kläser, A., Schmid, C., Cheng-Lin, L.: Action recognition by dense trajectories. In: CVPR. (2011)
7. Jiang, Y.G., Yang, J., Ngo, C.W., Hauptmann, A.G.: Representations of keypoint-based semantic concept detection: A comprehensive study. IEEE Trans. Multimedia **12**(1) (2010) 42–53
8. Merler, M., Huang, B., Xie, L., Hua, G., Natsev, A.: Semantic model vectors for complex video event recognition. IEEE Trans. Multimedia **14**(1) (2012) 88 –101
9. Natarajan, P., et al.: Bbn viser trecvid 2011 multimedia event detection system. In: NIST TRECVID Workshop. (2011)
10. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR. (2009)
11. Ferrari, V., Zisserman, A.: Learning visual attributes. In: NIPS. (2007)
12. Wang, Y., Mori, G.: A discriminative latent model of object classes and attributes. In: ECCV. (2010)
13. Lampert, C., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR. (2009)
14. Siddiquie, B., Feris, R., Davis, L.: Image ranking and retrieval based on multi-attribute queries. In: CVPR. (2011)
15. Liu, J., Kuipers, B., Savarese, S.: Recognizing human actions by attributes. In: CVPR. (2011)
16. Gaidon, A., Harchaoui, Z., Schmid, C.: Actom sequence models for efficient action detection. In: CVPR. (2011)
17. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: ECCV. (2010)
18. Tang, K., Fei-Fei, L., Koller, D.: Learning latent temporal structure for complex event detection. In: CVPR. (2012)
19. Laptev, I.: On space time interest points. IJCV **64** (2005)
20. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features (2005) IEEE International Workshop on VS-PETS.
21. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. IJCV **42** (2001) 145–175
22. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60**(2) (2004)
23. Rabiner, L., Juang, B.H.: Fundamentals of Speech Recognition. Prentice Hall (1993)
24. Do, T.M.T., Artières, T.: Large margin training for hidden markov models with partially observed states. In: ICML. (2009)
25. : Trecvid multimedia event detection track. "http://www.nist.gov/itl/iad/mig/med11.cfm" (2011)
26. Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88** (2010) 303–338